Color Correction and Depth Based Hierarchical Hole Filling in Free Viewpoint Generation

Yiguo Qiao, Licheng Jiao, Shuyuan Yang, Biao Hou and Jie Feng

***Abstract*—A high quality DIBR based free viewpoint generation model is proposed in this paper. Firstly, a boundary-aware 3D warping is introduced to project the side views from the side view plane to the virtual view plane. During the warping, a high-pass (HP) filter is used to remove the ghost contours around the object boundaries. Secondly, a color correction based image blending (CC-IB) procedure is put forward for synthesizing the two warped side views. In this CC-IB process, the disoccluded regions are color-corrected in the first place, based on an equivalent substitution and a polynomial fitting method. By blending the color-corrected disocclusions and the un-occluded regions, a color-continuous blended virtual view is produced. Lastly, a depth guided hierarchical hole-filling (D-HHF) is presented to fill the holes in the blended virtual view. With the assistance of depth, holes are divided into two kinds and only background pixels will be selected for filling the holes. The proposed D-HHF not only results in clear textures but also executes in a fast speed. Experimental results show that the proposed method can produce high quality virtual views with comfortable visual perceptions. Comparisons with the other methods demonstrate that the proposed method can achieve good performance in both visual quality and quantitative evaluation.**

***Index Terms*—DIBR, Free viewpoint generation, color correction, polynomial fitting, hole-filling.**

I. Introduction

3DTV is one of the most popular mass media in recent years, it can provide the audience with natural and vivid visual entertains[1][2][3][4][5]. FTV is regarded as the ultimate of 3DTV, which consists of infinite number of virtual views and enables the users to view 3D scenes by freely changing their viewpoints[6][7][8].

Free viewpoint generation has become one of the key technologies in FTV, which can be used in 3D video-conferences, 3D movies and so on [9][10]. Presently, sparse 3D model-based rendering (MBR) and depth-image-based-rendering (DIBR) are two common methods in virtual view synthesis[11][12]. DIBR is an core technique for stereo image generation[13][14][15], free viewpoint generation, the emerging area 3D unseen visible watermarking (UVW) scheme and so on. Compared with the traditional MBR view generation method, DIBR introduces the depth information into the view synthesis and do not need to reconstruct a 3D model. Basically, three processes are included in DIBR, pre-processing of depth maps, 3D warping, and hole-filling. Due to the simplicity and fast speed, DIBR has been widely used nowadays.

The DIBR based free viewpoint generation system makes use of the neighboring side views and their corresponding depth maps as input, i.e., the left view, the right view, the left depth map and the right depth map. Then through a series of processes, including the 3D warping, the image blending and the hole-filling, the system finally ouputs the synthesized virtual views[16].

A. Related work

Technological challenges exist in all the above three processes:

- During the 3D warping process, cracks and ghost contours will be brought in due to the round-off errors and the sharp changes of depths[17][18].
- Image blending is actually a synthesis of two side views. As the shooting environments and the camera configurations of the two side views may not exactly the same, color mismatch is a universal problem in the synthesized view[19][20][21].
- Lastly, holes are produced because of the still remaining occlusions and the incorrect depth values[22][23][24].

In view of the above challenges, lots of works have been done yet[18][25]. For removing the cracks that produced during the 3D warping, a median filter is commonly used on the warped depth maps. Several strategies have been put forward for removing the cracks in color views. The most direct way is to fill the cracks by copying the neighboring non-blank pixels based on a Z-buffer criterion. In 2010, Oh et al. proposed an inverse 3D warping on the color views to prevent the generation of cracks[21]. More specifically, Zarb et al. eliminate the cracks by using regular 3D warping on non-blank area and inverse 3D warping on blank area[26]. This approach greatly reduces the warping time compared with the complete inverse warping.

Ghost contours can be found at the object boundaries as Fig.1 shows, since pixels on the foreground-side boundaries may be warped to the background, these un-natural looking blurs are produced. One solution to remove the ghost contours is to expand the border of the disocclusions. Mori et al. proposed a boundary matting algorithm to remove the ghost contours[16], in which the disocclusions are dilated with one pixel wide. This method could not erase the ghost contours completely, since the ghost contours are usually three pixels wide. Certainly, one can dilate the occlusions with around a number of three pixels for erasing the ghost contours. However, not only the ghost contours but also the correct textures will be erased through this way. Another solution is to detect and dilate the depth edges first of all, the dilated regions will be omitted during the 3D warping later[18].

To avoid the color misalignment, Oh et al. proposed a histogram matching strategy[21][27], which maps the color of the side view to the virtual view. Nevertheless, color

discontinuities still remain between the disocclusions and the un-occluded regions. Yang et al. proposed a statistical method for removing the color discontinuities[28]. In their method, the brightness of one side view is used as the benchmark, and the brightness of other side view will be matched to this benchmark. Zarb et al. improved Yang's method by extracting the background regions and utilizing only background regions to determine the benchmark[26].

A large number of hole-filling algorithms have been proposed to remove the holes until now. The fast marching method (FMM) proposed by Telea et al. is a classical image in-painting method[29]. In order to fill the holes with background pixels, Oh et al. improved the FMM by preprocessing the boundary of the holes[21]. They replace the boundary pixels located in the foreground regions with the ones located in the background regions. Solh et al. presented a pyramid-like hierarchical hole-filling (HHF) method in 2010[30]. Yang et al. introduced a k-means clustering based hole-filling method, the k-means clustering is used to judge whether a hole belongs to occluded regions or not[28]. If belongs, only background pixels will be used for filling the hole with a asymmetric dilation approach. This method is highly depends on the k-means classification result.

B. Contributions

Main contributions of the proposed method are summarized as follows:

1) Before 3D warping, a threshold-controlled high-pass (HP) filter is designed for labeling the background-side depth boundaries. The labeled pixels will be omitted during the 3D warping, ghost contours can be prevented effectively in this way. Besides, as only background-side boundaries are detected, the foreground-side boundaries will be regularly warped and preserved.
2) An equivalent substitution is introduced to accomplish the color correction of the disoccluded areas. More specifically, the authors use the color difference between the disocclusion-surrounding areas in the side view and that in the virtual view, to replace the color difference between the disocclusions in the side view and that in the virtual view. With the proposed color correction strategy, color mismatches are effectively removed as Fig.1 shows.
3) A depth guided hierarchical hole-filling (D-HHF) is proposed for filling the holes with exact background pixels. And due to the hierarchical design, the D-HHF results in clear and accurate textures of the holes.

The rest of this paper is organized as follows. Section II presents the proposed method in detail. Experimental results and analysis are shown in Section III. Section IV concludes the paper.

II. Proposed Method

The proposed method mainly contains three portions: 1) boundary-aware 3D warping, 2) color correction based image blending (CC-IB), 3) D-HHF.

(a) (b)

Fig. 1. Final synthesized virtual view with (a) free viewpoint generation with regular 3D warping and image blending, and (b) free viewpoint generation with the proposed strategies.

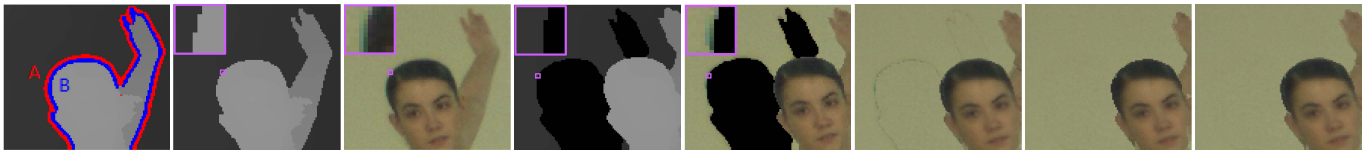
A. Boundary-aware 3D Warping

Due to the sharp changes of depths along the boundary regions, errors looks like ghost contours will be brought in during the traditional 3D warping[31]. Fundamentally speaking, this might be traced to the special capturing system of depth maps, which relying on the depth sensors. Generally, a depth sensor emits a certain range of modulated light at a known frequency and measures the phase shift between the transmitting and reverse signals. The depth is calculated based on the above phase shift, and this results in the sharp changes of depths along the boundaries. However, the boundary regions are usually mixed in a color image due to the pinhole imaging principle.

More specifically, the boundary regions in a depth map, as Fig.2(b) shows, can be definitely decomposed into a background-side border A and a foreground-side border B as marked in Fig.2(a). However, the foreground pixels and the background pixels on the two borders are infiltrated into each other in the color image as Fig.2(c) shows. With the regular 3D warping, depth pixels on background-side border A and foreground-side border B are accurately fall into the background and the foreground, respectively, as Fig.2(d) shows. However, the color pixels on border A and border B are still mixed as Fig.2(e) shows. The mixed pixels located on the background will further lead to ghost contours as illustrated in Fig.2(f).

In order to remove the ghost contours appeared in the background regions, image boundaries are detected and dilated firstly in some modified 3D warping methods, and these regions will be omitted during the 3D warping process. In other words, not only pixels on the background-side border A but also the ones on the foreground-side border B are failed to be warped, which further results in the erosion of the foreground areas as Fig.2(g) shows.

The proposed boundary-aware 3D warping contributes to both the prevention of the ghost contours on border A side and the protection of the original boundary on border B side, as demonstrated in Fig.2(h). Two steps are included in the proposed boundary-aware 3D warping, the background-side boundary detection and the detected boundary-aware 3D warping.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 2. Illustration of the proposed boundary-aware 3D Warping. (a) labeling the background-side border and the foreground-side border, (b) reference depth map, (c) reference color image, (d) depth map after 3D warping, (e) color image after 3D warping, (f) final synthesized virtual view with classical 3D warping, (g) final synthesized virtual view with boundary dilation based 3D warping, (h) final synthesized virtual view with the proposed boundary-aware 3D warping.

1) Background-side boundary detection: In the regular 3D warping, the generated ghost contours at border A are usually three pixels wide, which present as a belt-like area. For detecting that belt-like area, a threshold controlling HP filter is designed as shown in Eq.(1),

$$\sum_{m=-w}^{w}\sum_{n=-w}^{w} D(i+m,j+n) - a * D(i,j) > d_{th} \quad (1)$$

where D denotes the reference depth map; (i,j) denotes the pixel coordinate; w denotes the radius of the filtering window; a is a scale parameter which can be set as $((2w+1)^2-1)$; d_{th} is a given depth threshold.

With the proposed HP filter, only background-side boundary at border A will be detected, and the foreground-side boundary at border B will be skipped as expected. Besides, one can control the width of the detected area by adjusting the filter radius w, so that the detected area is never too broad or too narrow and can just cover the undesired ghost contours.

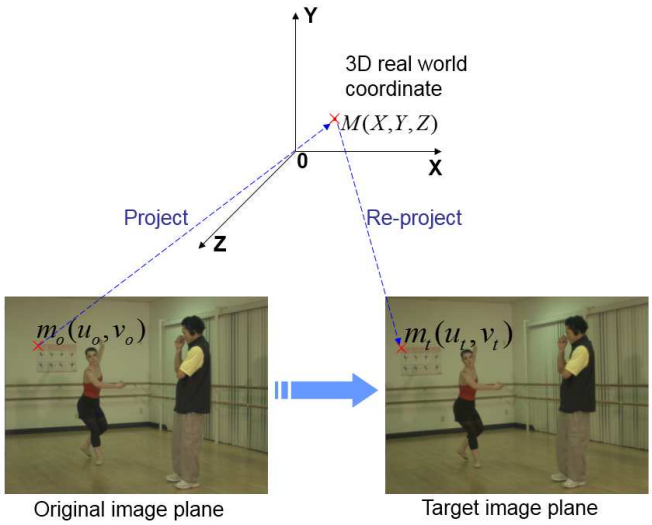


Fig. 3. Illustration of the 3D warping.

2) 3D warping: The proposed boundary-aware 3D warping is conducted on the reference depth maps, in which the above detected background-side boundaries will be omitted and unwarped, the rest regions will be warped by the regular 3D warping.

3D warping is used to project the views from the original view plane to the target virtual view plane[20][32]. Fig.3 provides a graphical illustration of the 3D warping process. Firstly, points on the original view plane will be projected to the 3D real world. Secondly, projected points in the 3D real world will be re-projected to the target virtual view plane[16].

The above projection and re-projection processes are formulated as in Eq.(2),

$$\begin{cases} P_o M = s_o m_o \\ P_t M = s_t m_t \end{cases} \quad (2)$$

where $m_o = [u_0, v_0, 1]^T$, $m_t = [u_t, v_t, 1]^T$ stand for the pixels in the original view plane and the target virtual view plane, respectively; $M = [X, Y, Z, 1]^T$ represents the points in the 3D real world; s_o and s_t are two non-zero scalars; P_o and P_t are projection matrixes of the original view plane and the target virtual view plane respectively. A projection matrix P can be decomposed as Eq.(3),

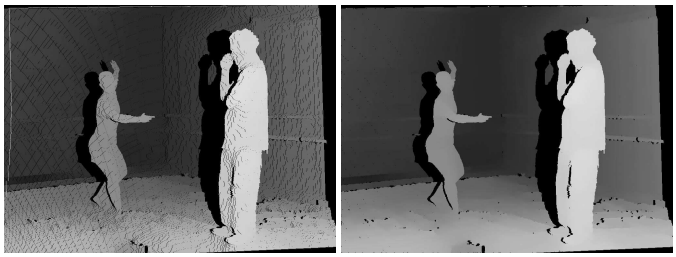
$$P = K[R; -Rt] \quad (3)$$

where K is the camera intrinsic matrix and $[R; -Rt]$ is the camera extrinsic matrix; Z denotes the depth of the observation point in the 3D real world and can be calculated as follows:

$$Z = \frac{1}{d/225\,((1/MinZ) - (1/MaxZ)) + (1/MaxZ)} \quad (4)$$

where d is the depth value in the depth map; $MinZ$ and $MaxZ$ are the nearest and the farthest depth in the real world, respectively.

During the 3D warping, several pixels in the original view plane may be projected into a same location in the virtual view plane. In this case, the pixel which owns the smallest depth will be selected as the original pixel, since only the nearest point from the viewer is visible. Eq.(5) formulates the



(a) (b)

Fig. 4. 3D warping and post-processing on the left side views. (a) 3D warped depth map and (b) filtered depth map.

(a) (b)

Fig. 5. Inverse warping on the color images. (a) warped left view and (b) warped right view.

(a) (b)

Fig. 6. Modification of the warped color images. (a) modified left view and (b) modified right view.

selection,

$$Z(u_t, v_t) = \arg\min_{O_{(u_0,v_0)}} Z(u_0, v_0) \tag{5}$$

where $O_{(u_0,v_0)}$ represents a pixel set in the original image plane, all pixels in this set will be projected in a same point (u_t, v_t) in the virtual image plane.

Undesirable cracks, which looks like black-contours, may be produced due to the round-off errors and the irregular depth changes. A median filter with a window size of $5*5$ is used to remove the cracks in the warped depth map[28]. Fig.4 provides the warped depth map and the filtered depth map from the left side.

Then an inverse warping is performed on the reference color images with the assistance of the filtered depth maps. As the name implies, inverse warping is the reverse operation of regular 3D warping. Firstly, non-zero points on the warped plane will be projected to the 3D real world, then projected points in the 3D real world will be re-projected to the reference view plane. Inverse warping of the color images can significantly prevent the cracks from occurring. Fig.5 shows the inverse warped color images.

B. Color Correction based Image Blending

It have been mentioned that color mismatches will be brought in during the image blending process, since the blended image is a synthesis of two side views. So that we introduce the CC-IB method to remove the color mismatches. The CC-IB is based on an equivalent substitution that the color difference between the occlusions of the side view and that of the virtual view, is almost the same as the color difference between the occlusion-surrounding regions of the side view and that of the virtual view. Four portions are included in the proposed CC-IB, firstly the un-occluded regions will be blended as a basis of the synthesized virtual view, secondly credible occlusion-surrounding regions will be extracted for the subsequent color correction, thirdly disocclusions will be reconstructed through the color correction of the occluded regions in the side views, lastly the virtual view is generated by synthesizing the un-occluded regions and the reconstructed disocclusions.

1) Un-occluded regions blending: In this section, un-occluded regions of the two warped side views will be blended. Before blending, the warped side views should be preprocessed to have the same holes as Fig.6 shows.

Then the modified left and right side views are blended as Eq.(6) shows,

$$B_{\_un}(u, v) = (1 - \alpha) \cdot M_L(u, v) + \alpha \cdot M_R(u, v) \tag{6}$$

where M_L and M_R are the modified left and right views respectively; α is a weight coefficient and can be computed as Eq(7) shows,

$$\alpha = \frac{|t - t_L|}{|t - t_L| + |t - t_R|} \tag{7}$$

t_L, t_R, and t are the translation vectors of the left camera, the right camera and the virtual camera, respectively.

The un-occlusion blended virtual image $B_{\_un}$ is shown in Fig.7, the color of which will be used as the basis color of the final synthesized virtual view.

2) Credible regions extraction: It is a reasonable assumption that, the occluded regions, i.e. the blank areas in Fig.7, must have similar color conditions with its surrounding regions. One step further, the color difference between the occluded regions of the side view and that of the virtual view must can be replaced with the color difference between the occlusion-surrounding regions of the side view and that of the virtual view.

More specifically, Fig.8 provides a visual descriptions of this equivalent substitution, in which, region A and region C are two occluded regions in $B_{\_un}$. Region A' in the warped

Fig. 7. Un-occlusion blended virtual image $B_{\_un}$.

(a) (b) (c) (d)

Fig. 8. Credible regions extraction. Details enlarged images of (a) occlusion blended virtual view $B_{\_un}$, (b) warped right view, (c) occlusion blended virtual view $B_{\_un}$ and (d) warped left view. Regions framed in the red line are the occluded regions, and framed in the blue line are their corresponding credible regions.

right view is the corresponding region of A, region C' in the warped left view is the corresponding region of C. Occlusions A and C can be filled with their corresponding regions A' and C', respectively. However, since $B_{\_un}$ is a blended view of the two side views, color differences may exist between region A and region A', region B and region B'. Based on the above equivalent substitution, color difference between regions A and A' can be replaced with the color difference between the surrounding regions B and B'. Similarly, color difference between regions C and C' can be replaced with the color difference between the surrounding regions D and D'.

Thus the credible surrounding regions B and D should be firstly extracted. As a example, the credible surrounding region B around the occluded region A can be obtained through a dilation operation as Eq.(8) and Eq.(9) shows,

$$B = B_{\_un} - (A \oplus S_e) \tag{8}$$

$$A \oplus S_e = \min_{(x,y)\in S_e} [A_{S_e}(x,y)] \tag{9}$$

where S_e is a disk-shaped structuring element which operates on A; A_{S_e} is a masked region with S_e and (x, y) is a pixel locates on S_e.

The disk radius of S_e various with different occluded regions, it can be empirically obtained as Eq.(10) shows,

$$r = 2 \cdot \ln[N(A)] \tag{10}$$

where $N(A)$ denotes the total number of pixels in A.

3) Disocclusions reconstructing: In this section, the disocclusions will be reconstructed by correcting the color of the occlusions of the side views. As mentioned, the authors use the color difference between the occlusion-surrounding regions of the side view and that of the virtual view to replace the color difference between the occluded regions of the side view and that of the virtual view. For this purpose, the polynomial fitting strategy is adopted. More specifically, a third-order polynomial transform is commonly used on the experience[33][34][35][36].

Reconstruction of the disoccluded region $\hat{A}$ from its corresponding region A', as an example, is detailed below. For the convenience of image storage and transmission, the proposed color correction is performed in YUV color space. Particularly, Y channel of the reconstructed disocclusion $\hat{A}$ can be fitted by a third-order polynomial of the YUV channels of the occluded region A', the fitting is expressed as Eq.(11) shows.

$$\begin{aligned} Y_{\hat{A}} =& C_{Y1}Y_{A'} + C_{Y2}U_{A'} + C_{Y3}V_{A'} + C_{Y4}Y^2_{A'} + C_{Y5}U^2_{A'} \\ &+ C_{Y6}V^2_{A'} + C_{Y7}Y_{A'}U_{A'} + C_{Y8}Y_{A'}V_{A'} + C_{Y9}U_{A'}V_{A'} \\ &+ C_{Y10}Y^3_{A'} + C_{Y11}U^3_{A'} + C_{Y12}V^3_{A'} + C_{Y13}Y_{A'}U_{A'}V_{A'} \\ &+ C_{Y14}Y^2_{A'}U_{A'} + C_{Y15}Y^2_{A'}V_{A'} + C_{Y16}Y_{A'}U^2_{A'} \\ &+ C_{Y17}Y_{A'}V^2_{A'} + C_{Y18}U^2_{A'}V_{A'} + C_{Y19}U_{A'}V^2_{A'} + C_{Y20} \end{aligned} \tag{11}$$

A simplification of the above expression is presented in Eq.(12),

$$Y_{\hat{A}} = \Psi_{A'} \cdot C_Y. \tag{12}$$

where $\Psi_{A'}$ is a combination matrix including different products of the three channels, which is defined as in Eq.(13); $C_Y = [C_{Y1}, C_{Y2}, ..., C_{Y20}]'$ is a 20×1 coefficient vector, which is used to map the Y channel from the occluded region A' to the disoccluded region $\hat{A}$. Equivalently, the coefficient vector of U channel C_U and the coefficient vector of V channel C_V can be similarly defined.

$$\begin{aligned} \Psi_{A'} =& [Y_{A'} \;\; U_{A'} \;\; V_{A'} \;\; Y^2_{A'} \;\; U^2_{A'} \;\; V^2_{A'} \;\; Y_{A'}U_{A'} \;\; Y_{A'}V_{A'} \\ & U_{A'}V_{A'} \;\; Y^3_{A'} \;\; U^3_{A'} \;\; V^3_{A'} \;\; Y_{A'}U_{A'}V_{A'} \;\; Y^2_{A'}U_{A'} \\ & Y^2_{A'}V_{A'} \;\; Y_{A'}U^2_{A'} \;\; Y_{A'}V^2_{A'} \;\; U^2_{A'}V_{A'} \;\; U_{A'}V^2_{A'} \;\; 1] \end{aligned} \tag{13}$$

Based on the assumption that the color difference between $\hat{A}$ and A' is approximately equal to the color difference between

(a) (b)

Fig. 9. Images of the reconstructed disocclusions from (a) the left side, and (b) the right side.

B and B', the Y channel of the credible surrounding region B can be presented as Eq.(14) shows,

$$Y_B = \Psi_{B'} \cdot C_Y \tag{14}$$

where $\Psi_{B'}$ is defined similar to $\Psi_{A'}$, which is shown in Eq.(15).

$$\begin{aligned}\Psi_{B'} =[&Y_{B'} \;\; U_{B'} \;\; V_{B'} \;\; Y^2_{B'} \;\; U^2_{B'} \;\; V^2_{B'} \;\; Y_{B'}U_{B'} \;\; Y_{B'}V_{B'} \\ &U_{B'}V_{B'} \;\; Y^3_{B'} \;\; U^3_{B'} \;\; V^3_{B'} \;\; Y_{B'}U_{B'}V_{B'} \;\; Y^2_{B'}U_{B'} \\ &Y^2_{B'}V_{B'} \;\; Y_{B'}U^2_{B'} \;\; Y_{B'}V^2_{B'} \;\; U^2_{B'}V_{B'} \;\; U_{B'}V^2_{B'} \;\; 1].\end{aligned} \tag{15}$$

From Eq.(14) and based on the least squares regression model, the coefficient vector C_Y can be calculated as Eq.(16) shows.

$$C_Y = (\Psi^T_{B'}\Psi_{B'})^{-1}\Psi^T_{B'}Y_B \tag{16}$$

By substituting Eq.(16) into Eq.(12), the color-corrected $Y_{\hat{A}}$ is finally obtained as shown in Eq.(17).

$$Y_{\hat{A}} = \Psi_{A'}(\Psi^T_{B'}\Psi_{B'})^{-1}\Psi^T_{B'}Y_B \tag{17}$$

The other two channels U and V can be calculated equivalently. Till now, the disoccluded region $\hat{A}$ is finally reconstructed.

The other disocclusions are reconstructed through the same way one after another. Total number of the occluded regions decides the computation costs of the polynomial fitting based color correction. For time saving and without debasing the image quality, regions with a fairly small number of pixels (say less than 50), will be ignored. Images of the reconstructed disocclusions are shown in Fig.9.

4) Image blending: The blended virtual image can be obtained by combining the un-occlusions and the reconstructed disocclusions, as Eq.(18) shows,

$$Bv = B_{\_un} + Dis_{\_L} + Dis_{\_R} \tag{18}$$

where $B_{\_un}$ denotes the un-occlusion blended virtual image, as mentioned previous; $Dis_{\_L}$ and $Dis_{\_R}$ denote the disoccluded images from the left side and the right side, respectively.

The final blended image is shown in Fig.10, which proves that the proposed CC-IB completely eliminate the color mismatches between the two side views. However, holes caused by remaining occlusions and incorrect depth values are still can be found. Thus a D-HHF method is proposed to remove those unexpected holes. Note that, since depth is of great use in the proposed D-HHF, the blended depth map Bd should also be generated accompany with the blended color image.

Fig. 10. Final blended virtual view B_V.

C. D-HHF

1) Review of HHF: The proposed D-HHF is inspired by the classical HHF method[30]. The HHF is composed of three parts: reduction, expansion and hole-filling.

- Reduction: By down-sampling the initial hole-remaining image R_0 in turn, we obtain the reduced sequence R_1, R_2, $\cdots R_N$. R_1 is the down-sampled version of R_0, R_2 is the down-sampled version of R_1, and so on. Through a pseudo Gaussian plus zero elimination filtering operation, both the image resolution and the holes are decreased. The reduction stops until no visible holes can be found.
- Expansion: From the lowest resolution hole-free image R_N, we interpolate it to a higher resolution version E_{N-1}. The expanded image E_{N-1} has a same size as R_{N-1}.
- Hole-filling: Fill the holes in R_{N-1} with the corresponding pixels in E_{N-1}, and get the hole-filled version F_{N-1}.

Repeat the expansion and the hole-filling processes until the highest resolution hole-filled image F_0 is obtained.

Since in the blended virtual view Bv, two kinds of holes exist. First, holes caused by the remaining occlusions are almost along the depth boundaries, which means, both the foreground and the background pixels are around the holes. In this case, using the background pixels rather than the foreground ones to fill the holes is more reasonable. Otherwise, holes caused by the incorrect depth values may locate in either foreground areas or background areas. In this case, we can fill the holes with their surrounding pixels in simple. Accordingly, we proposed the following D-HHF method.

2) Proposed D-HHF: The proposed D-HHF follows the framework of the classical HHF in general, specially, depth information is considered in both the reduction and the expansion processes. Framework of the proposed D-HHF is shown in Fig. 11.

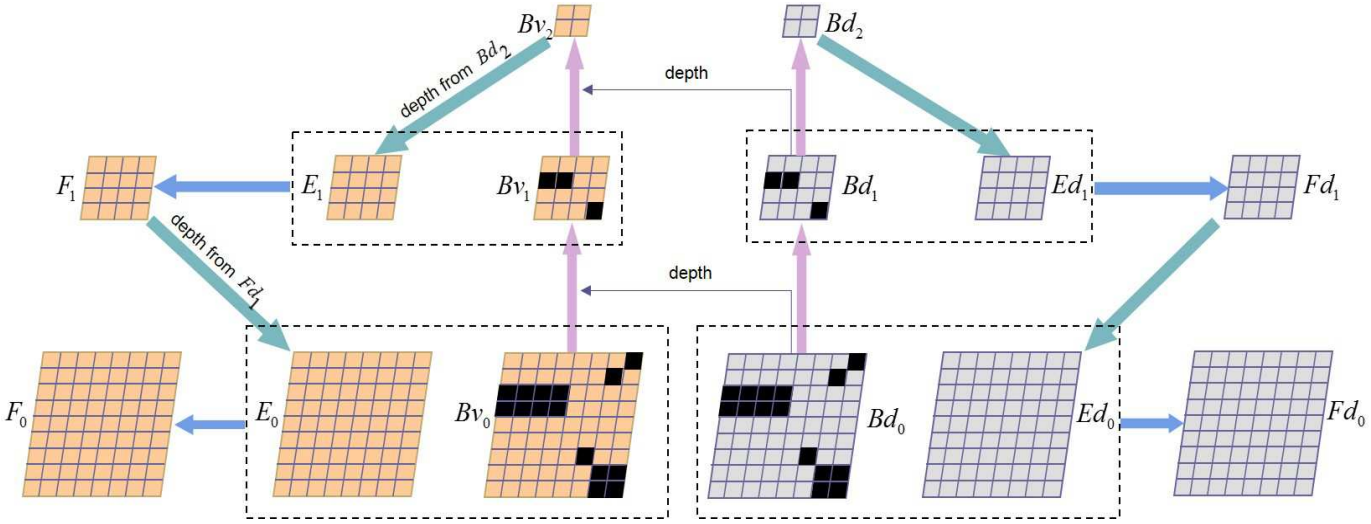


Fig. 11. Framework of the proposed D-HHF. Pink arrow represents the depth guided reduction process, green arrow represents the depth guided expansion process, blue arrow represents the hole-filling process.

- Depth guided reduction: Depth map plays an important role in this stage. On the one hand, it is used to label different kinds of holes as mentioned above, thus different filtering patterns will be adopted according to different kinds of holes. On the other hand, depth map is used to select the significant pixels for reduction, generally speaking, only pixels belonging to background will be selected. One step further, depth is introduced as the guidance of the color image reduction, and this depth guided reduction establishes the basis for filling the holes with background pixels. As color image is reduced layer by layer, depth map worked as a guide should also be reduced simultaneously with the color image. So that both the reduced depth sequence $Bd_1,\ Bd_2,\ \cdots Bd_N$ and the reduced color sequence $Bv_1,\ Bv_2,\ \cdots Bv_N$ are calculated. More specifically, the depth version Bd_0 not only guides the reduction of itself, but also guides the reduction of the color image Bv_0.
- Depth guided expansion: Depth map is also introduced as the guidance of the expansion process, which is used to select the background pixels to accomplish the up-sampling. Thus depth map is also expanded each layer along with the expansion of the color image, similar as that in the above depth guided reduction process. More exactly, Bd_N guides the expansion of itself to Ed_{N-1}, and it also guides the expansion of the color image Bv_N to E_{N-1}.
- Hole-filling: Both the holes in the depth map and the holes in the color image are filled in this stage. Holes in the depth map Bd_{N-1} are filled by corresponding pixels in Ed_{N-1}, and thus leads the hole-filled depth map Fd_{N-1}. Holes in the color image Bv_{N-1} are filled with the corresponding pixels in E_{N-1}, and thus results in the hole-filled color image F_{N-1}.

By repeating the expansion and the hole-filling processes alternately, the hole-filled depth maps $Fd_{N-1},\ Fd_{N-2},\ \cdots Fd_0$ and the hole-filled color views $F_{N-1},\ F_{N-2},\ \cdots F_0$ are generated in turn. Details of the depth guided reduction and the depth guided expansion are introduced below.

$$\begin{cases} Bd_{k+1} = Reduce[Bd_k, Bd_k] \\ Bv_{k+1} = Reduce[Bv_k, Bd_k] \end{cases} \tag{19}$$

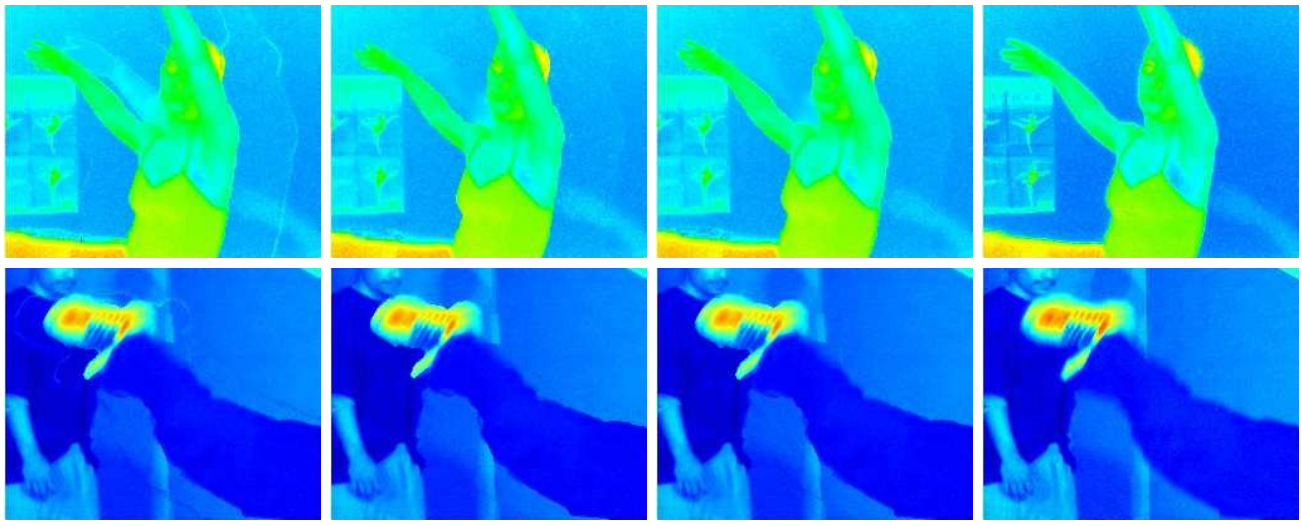
$$N(i) = \begin{cases} 1 & i \geqslant 0 \\ 0 & else \end{cases} \tag{21}$$

As mentioned, holes should be labeled into different kinds firstly, according to the depths of the surrounding pixels. Specifically, in a depth version Bd_k, a hole surrounded by both foreground pixels and background pixels will be labeled by 1, a hole surrounded by foreground pixels or background pixels alternatively will be labeled by 2, other non-hole areas will be labeled by 0, and this results in a labeled graph L_k.

A pseudo Gaussian plus average filtering is adopted to accomplish the depth guided reduction, which is formulated as in Eq.(19) shows, where $Reduce$ represents a block-based down-sampling function. For definiteness and without loss of generality, we formulate the $Reduce$ process on the color image as Eq.(20) shows, where $X_{m,n}$, $Y_{m,n}$ and $Z_{m,n}$ are block windows in Bv_k, Bd_k and L_k, respectively, both of them are located on their top-left point (m, n); s denotes the side length of the block window; qh is a depth threshold for dividing the background pixels from the foreground ones; nz denotes the total number of the non-zero pixels; N is a step function shows in Eq.(21). The reduction of the depth map is equivalent to that of the color image.

Then the reduction of a current block can be divided into four cases as shown in Eq.(20). In the first case, the current block does not cover any hole pixel, and in this case we generate the down-sampled pixel by using a gaussian filter. In the second case, the current block covers a hole which surrounded by both background pixels and foreground pixels, and in this case we use an average filter on only background pixels (if any) to generate the down-sampled pixel. In the third

$$Bv_{k+1}[m,n]=\begin{cases}\sum_{i=1}^{s^2} g(i)X_{m,n}(i), & all(Z_{m,n}=0)\\ \frac{\sum_{i=1}^{s^2} X_{m,n}(i)N(qh-Y_{m,n}(i))}{\sum_{i=1}^{s^2} N(qh-Y_{m,n}(i))}, & any(Z_{m,n}=1)\&\sum_{i=1}^{s^2} N(qh-Y_{m,n}(i))>0\\ \frac{\sum_{i=1}^{s^2} X_{m,n}(i)}{nz}, & any(Z_{m,n}=2)\&nz>0\\ 0, & else\end{cases} \tag{20}$$



(a) (b) (c) (d)

Fig. 12. 3D warping results of top: $Ballet$ and bottom: $Breakdancers$ with different methods. Magnified details of the 3D warping results based on (a) the based on the regular 3D warping, (b) the dilation based 3D warping, (c) the proposed boundary-aware 3D warping and (d) the ground truth.

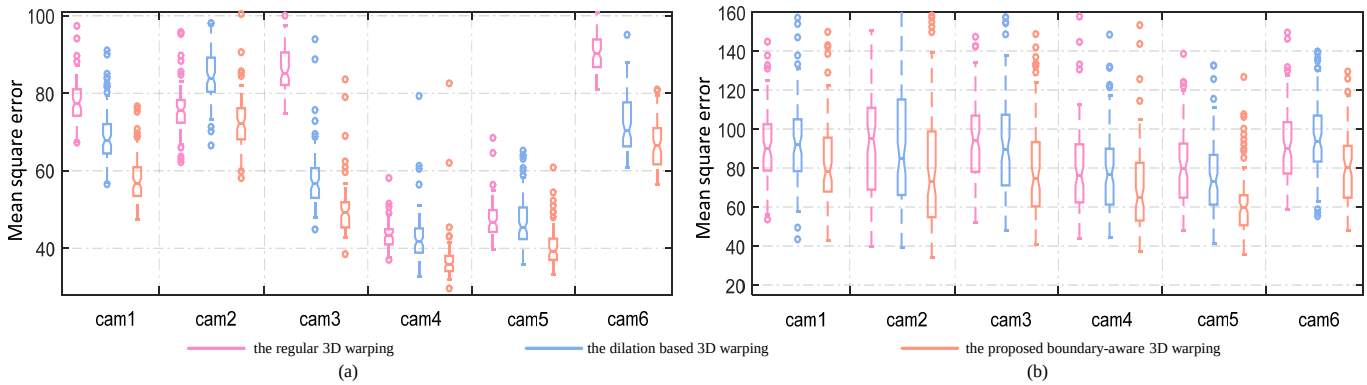
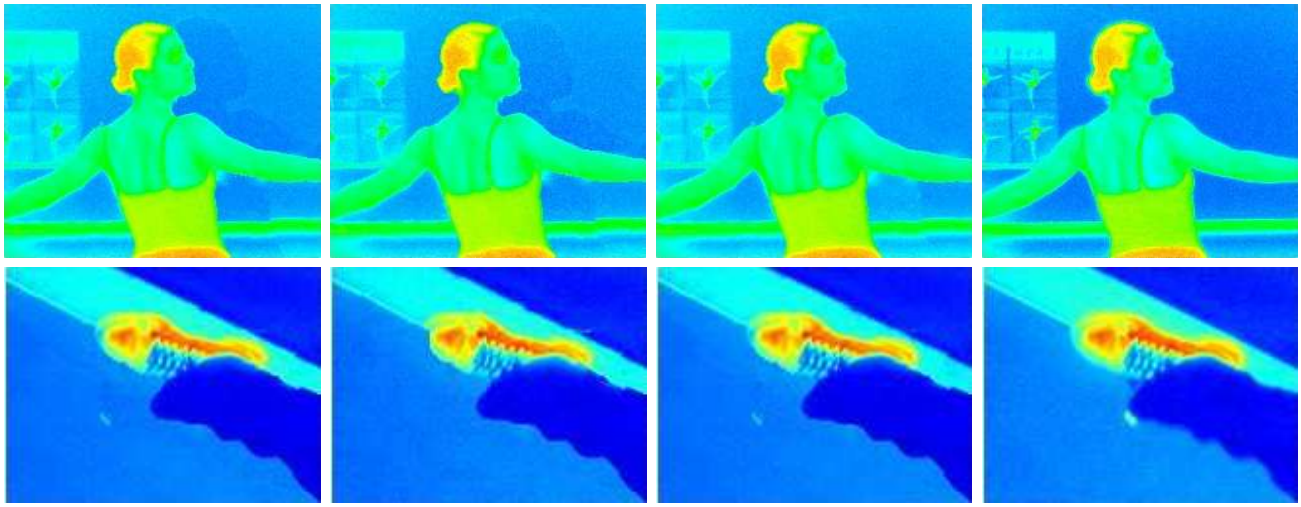


Fig. 13. Quantitative evaluations of different 3D warping methods on (a) $Ballet$ and (b) $Breakdancers$. The horizontal axis represents different viewpoints and the vertical axis shows the MSE of the boundary regions of all the 100 frames of each viewpoint. The pink, blue and orange markers represent the regular 3D warping, the dilation based 3D warping and the proposed boundary-aware 3D warping, respectively.



(a) (b) (c) (d)

Fig. 14. Virtual image blending results of top: $Ballet$ and bottom: $Breakdancers$ with different methods. Magnified details of the blended virtual views based on (a) the regular image blending, (b) the histogram based image blending, (c) the proposed CC-IB and (d) the ground truth.

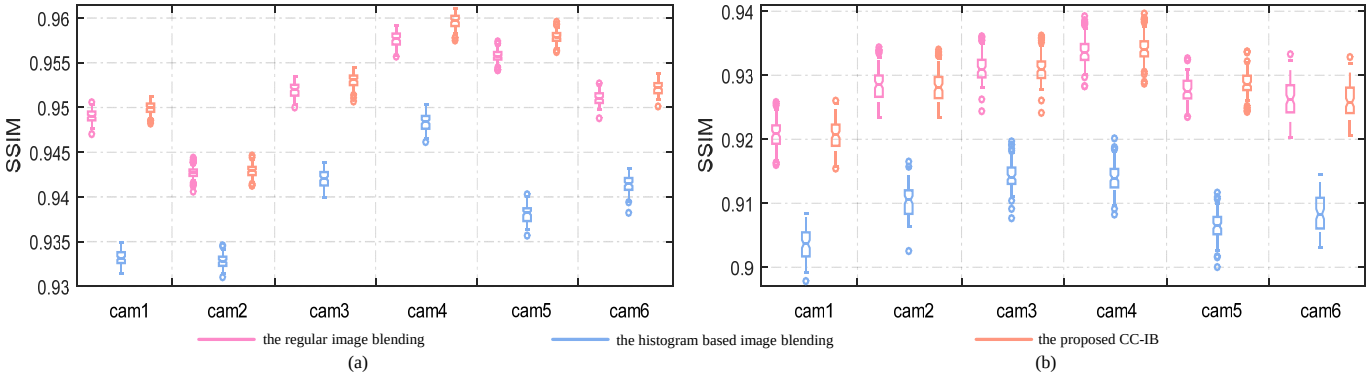


Fig. 15. Quantitative evaluations of different image blending methods on (a) $Ballet$ and (b) $Breakdancers$. The horizontal axis represents different viewpoints and the vertical axis shows the SSIM of all the 100 frames of each viewpoint. The pink, blue and orange markers represent the regular image blending, the histogram based image blending and the proposed CC-IB, respectively.

case, the current block covers a hole which surrounded by background pixels or foreground pixels alternatively, and in this case we use an average filter on all non-zero pixels (if any) to generate the down-sampled pixel. Otherwise, we set the down-sampled pixel as 0. Based on this strategy, background information can be largely used in filling the holes.

The depth guided expansion is formulated as in Eq.(22) shows,

$$\begin{cases} Ed_k = Expand[Ed_{k+1}, Ed_{k+1}] \\ E_k = Expand[E_{k+1}, Ed_{k+1}] \end{cases} \tag{22}$$

where $Expand$ is a up-sampling function contrary to the $Reduce$. It is formulated as in Eq.(23) shows, where t denotes the search range; $(m+i)/2$ and $(n+j)/2$ should be integers. The depth map is expanded equivalently. It shows that the depth guided expansion is processed as an average filtering. Once there has background pixels within a certain search range, the current pixel will be interpolated by the mean of the background pixels. Otherwise, all surrounding pixels of the current pixel are belonging to the foreground, and in this case, the current pixel will be interpolated by the mean of foreground pixels.

III. Experimental Results

A. Experimental setup

The experiments are performed on a PC with Core Duo 3.2GHz CPU and 4G RAM, by using Matlab based on the Window 7 operation system. Five data sets are used in the experiments, including *Ballet* and *Breakdancing* provided by Microsoft Research, *Kendo* and *Balloons* provided by Nagoya, and *Lovebird1* provided by ETRI/MPEG Korea Forum[37]. *Breakdancing* and *Ballet* are captured from 8 different viewpoints, and each viewpoint contains 100 frames. *Kendo* and *Balloons* are captured from 7 different viewpoints, and each viewpoint contains 300 frames. *Lovebird1* is captured from 12 different viewpoints, and each viewpoint contains 300 frames. All these data sets owns an image resolution of $1024 * 768$.

In the experiments, for *Breakdancing* and *Ballet*, synthesized virtual views from viewpoint 1 to 6 are tested, and each virtual view are synthesized by its left side view and right side view. For *Kendo* and *Balloons*, virtual views of viewpoint 3 are tested, which are synthesized by views from viewpoint 1

$$E_k[m,n]=\begin{cases}\dfrac{\sum_{i=-t}^{t}\sum_{j=-t}^{t}E_{k+1}[\frac{m+i}{2},\frac{n+j}{2}]\cdot N(qh-Ed_{k+1}[\frac{m+i}{2},\frac{n+j}{2}])}{\sum_{i=-t}^{t}\sum_{j=-t}^{t}N(qh-Ed_{k+1}[\frac{m+i}{2},\frac{n+j}{2}])}, & \sum_{i=-t}^{t}\sum_{j=-t}^{t}N(qh-Ed_{k+1}[\frac{m+i}{2},\frac{n+j}{2}])\neq 0\\ \dfrac{\sum_{i=-t}^{t}\sum_{j=-t}^{t}E_{k+1}[\frac{m+i}{2},\frac{n+j}{2}]\cdot N(Ed_{k+1}[\frac{m+i}{2},\frac{n+j}{2}]-qh)}{\sum_{i=-t}^{t}\sum_{j=-t}^{t}N(Ed_{k+1}[\frac{m+i}{2},\frac{n+j}{2}]-qh)}, & else\end{cases} \tag{23}$$

and 5. For *Lovebird1*, virtual views of viewpoint 7 are tested, which are synthesized by views from viewpoint 5 and 8[1][38]. Moreover, for definiteness and without loss of generality, all frames of each viewpoint are tested.

B. Evaluations of each sub-step

Three tests are set up according to the three sub-steps, i.e. the boundary-aware 3D warping, the CC-IB and the D-HHF. Besides, synthesized virtual views under 6 different viewpoints on sequences *Breakdancing* and *Ballet* are used for evaluation. Experimental results are provided from Fig.12 $\sim$ Fig.17, rather, Fig.12, Fig.14 and Fig.16 present the visual results of the three sub-steps, respectively. And the quantitative evaluations are provided in Fig.13, Fig.15 and Fig.17, respectively.

The proposed boundary-aware 3D warping is compared with the regular 3D warping[16] and the dilation based 3D warping[18], visual results and quantitative evaluations are provided in Fig.12 and Fig.13, respectively. Fig.12 shows that the proposed boundary-aware 3D warping not only removes the ghost contours produced by the regular 3D warping, but also avoids the erosion of the foreground brought by the dilate based 3D warping. To evaluate the boundary-aware 3D warping quantitatively, mean square error (MSE) of boundary regions is utilized since the improvements of this process are completely in boundary regions. Results in Fig.13 also verifies the superiority of the boundary-aware 3D warping.

The proposed CC-IB is compared with the regular image blending and the histogram based image blending, Fig.14 and Fig.15 present the visual results and the quantitative evaluations, respectively. Since CC-IB makes contributions in removing the color discontinuities, the structural similarity (SSIM) is introduced to evaluate the performance of CC-IB. Different data sets of *Breakdancing* and *Ballet* indicate that the larger the color differences between the side views, the greater the benefits of the proposed CC-IB.

Visual results of the proposed D-HHF are shown in Fig.16, the method is compared with the classical HHF method[30], the FMM method[29] and the depth based FMM method[21]. Comparisons show that holes are filled more clearly and accurately based on the proposed D-HHF. Moreover, MSE of the hole regions is used for quantitative evaluations, results in Fig.17 demonstrate the validity of the D-HHF.

C. Evaluations of the proposed model

To evaluate the performance of the proposed free viewpoint generation model, it is compared with some other state-of-art models, including the original view synthesis model[16], the histogram matching based view synthesis model[21], view synthesis model in [18] and VSRS (4.2)[39]. Visual results are shown in Fig.18, which indicate that synthesized virtual views based on the proposed model are both color-continuous and natural-looking.

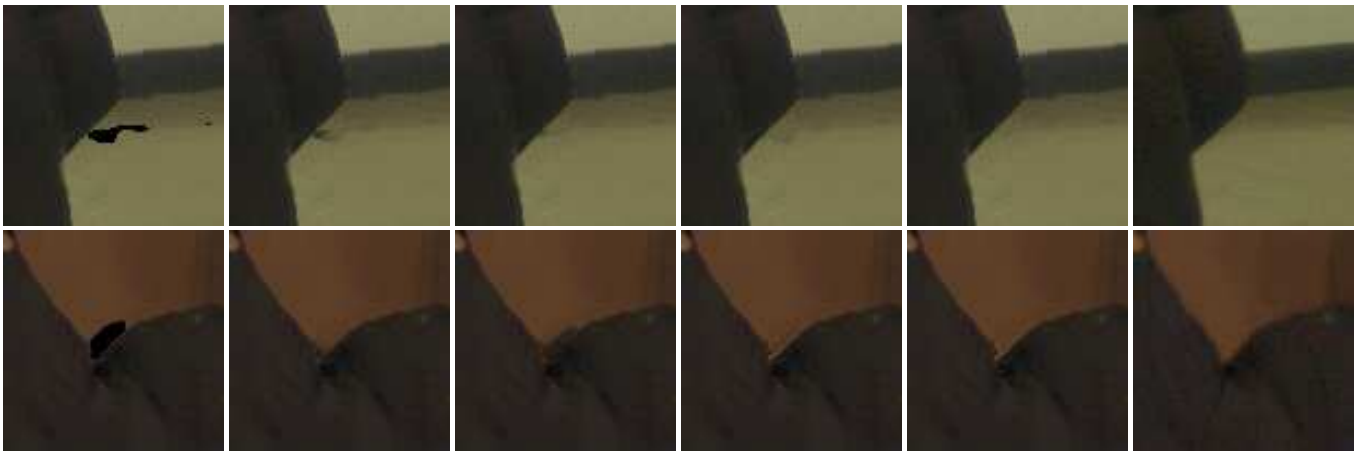
Moreover, quantitative evaluations of both objective and subjective are conducted. Peak signal-to-noise ratio (PSNR) and SSIM[40] are used for objective evaluations, evaluation results are presented in Table I. One thing to note is that in capturing *Breakdancing* and *Ballet*, the camera rotation matrixes of different viewpoints are not exactly the same, however, VSRS takes an equal camera rotation matrix as a default setting, which further results in unsatisfactory results of VSRS on these two data sets. Besides, more to explain is why the proposed method does not improve that much in PSNR and SSIM evaluations. That is because PSNR and SSIM are evaluations of the entire image, whereas the proposed method focuses on improving some local regions, including the boundaries, the occlusions and the holes. Image quality of such regions greatly affects the visual perception, and for this reason, subjective evaluations are conducted.

The subjective evaluations are implemented on the platform of the MSU Perceptual Video Quality tool[1]. The DSCQS (Double Stimulus Continuous Quality Scale) type II, which is relaying on the ITU-R BT.500, is selected. Moreover, 50 observers, including 15 Ph.D. students, participated in the ratings. Scores from 1 to 5 are on behalf of very annoying, annoying, slightly annoying, perceptible, imperceptible, respectively. Average scores of the 50 observers are provided in Table II.

IV. Conclusion

In this paper, we propose a high quality DIBR based free viewpoint generation model. In which, a boundary-aware 3D warping is introduced to remove the ghost contours firstly. Then a CC-IB is introduced to generate the blended virtual view, by correcting the color of the occluded regions with the color of the corresponding occlusion-surrounding regions, the

[1]http://www.compression.ru/video/quality_measure/perceptual_video_quality_tool_en.html



(a) (b) (c) (d) (e) (f)

Fig. 16. Hole-filling results of top: $Ballet$ and bottom: $Breakdancers$ with different methods. Magnified details of (a) regions with holes and the final hole-filling results based on (b) the HHF, (c) the FMM, (d) the depth based FMM, (e) the proposed D-HHF and (f) the ground truth.

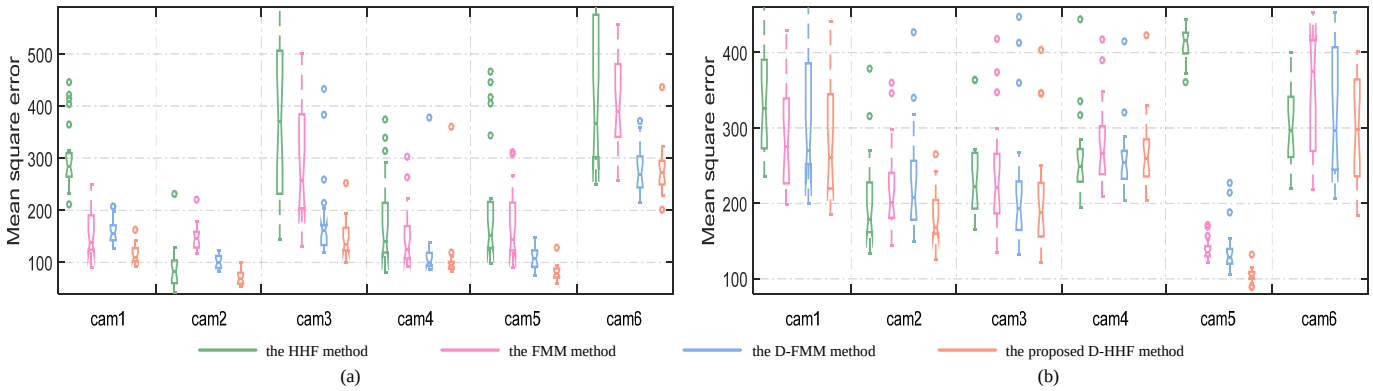
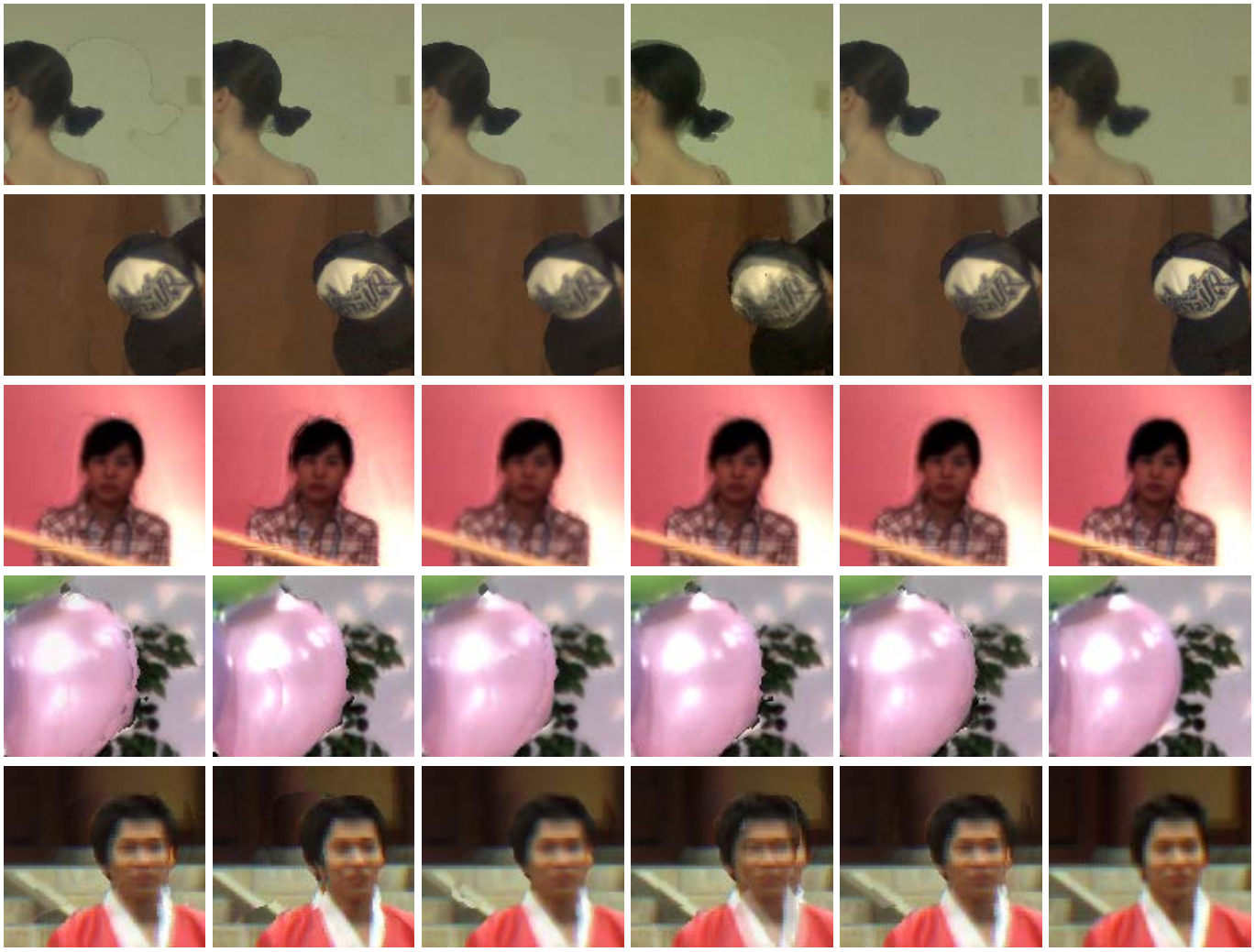


Fig. 17. Quantitative evaluations of different hole-filling methods on (a) $Ballet$ and (b) $Breakdancers$. The horizontal axis represents different viewpoints and the vertical axis shows the MSE of the hole regions of all the 100 frames of each viewpoint. The green, pink, blue and orange markers represent the classical HHF, the FMM, the depth based FMM and the proposed D-HHF, respectively.

TABLE I
PSNR AND SSIM EVALUATIONS(ALL FRAMES OF EACH VIRTUAL VIEWPOINT ARE EVALUATED).

Seq.		*Ballet*			*Breakdancers*			*Kendo*	*Balloons*	*Lovebird*1
Syn.		0→1←2	2→3←4	4→5←6	0→1←2	2→3←4	4→5←6	1→3←5	1→3←5	5→7←8
PSNR	[16]	31.04±0.16	31.92±0.13	32.85±0.12	31.94±0.34	33.10±0.29	33.68±0.28	33.73±0.10	30.56±0.11	32.03±0.18
	[21]	30.11±0.12	31.47±0.11	31.75±0.13	31.39±0.29	32.12±0.28	32.30±0.30	31.39±0.08	28.85±0.09	32.89±0.16
	[18]	31.09±0.15	**32.04**±0.13	32.52±0.12	**32.16**±0.39	33.11±0.30	**33.98**±0.29	33.52±0.09	30.76±0.08	32.16±0.17
	VSRS	28.19±0.19	29.45±0.16	29.72±0.16	29.42±0.45	29.67±0.44	29.66±0.47	34.66±0.07	31.53±0.06	32.29±0.14
	Pro.	**31.32**±0.13	32.02±0.12	**33.57**±0.09	32.11±0.37	**33.17**±0.34	**33.98**±0.30	**34.92**±0.08	**31.58**±0.07	**33.01**±0.15
SSIM (±%)	[16]	0.9458±0.10	0.9484±0.09	0.9532±0.08	0.9174±0.26	0.9289±0.25	0.9247±0.21	0.9601±0.07	0.9448±0.07	0.9392±0.09
	[21]	0.9317±0.08	0.9416±0.09	0.9380±0.10	0.9025±0.29	0.9138±0.24	0.9061±0.18	0.9432±0.06	0.9191±0.06	0.9081±0.07
	[18]	0.9472±0.07	0.9509± 0.09	0.9537± 0.08	**0.9229**±0.26	**0.9322**±0.22	0.9286±0.19	0.9613±0.07	0.9478±0.06	0.9462±0.08
	VSRS	0.9017±0.10	0.9113±0.10	0.9127±0.09	0.8910±0.26	0.8982±0.21	0.8857±0.23	0.9602±0.05	0.9559±0.05	0.9510±0.06
	Pro.	**0.9498**±0.06	**0.9530**±0.08	**0.9579**±0.07	0.9208±0.27	0.9311±0.22	**0.9289**±0.19	**0.9615**±0.05	**0.9561**±0.06	**0.9591**±0.07



(a) (b) (c) (d) (e) (f)

Fig. 18. Magnification of the final synthesized virtual views based on (a) the original view synthesis model[16], (b) the histogram matching based view synthesis model[21], (c) view synthesis model in [18], (d) VSRS (4.2)[39] and (e) the proposed method, and (f) the ground truth. From top to bottom are the result on data sets *Ballet*, *Breakdancing*, *Kendo*, *Balloons* and *Lovebird1*, respectively.

TABLE II
SUBJECTIVE EVALUATIONS(ALL FRAMES OF EACH VIRTUAL VIEWPOINT ARE EVALUATED).

Seq.		*Ballet*			*Breakdancers*			*Kendo*	*Balloons*	*Lovebird*1
Syn.		$0\rightarrow1\leftarrow2$	$2\rightarrow3\leftarrow4$	$4\rightarrow5\leftarrow6$	$0\rightarrow1\leftarrow2$	$2\rightarrow3\leftarrow4$	$4\rightarrow5\leftarrow6$	$1\rightarrow3\leftarrow5$	$1\rightarrow3\leftarrow5$	$5\rightarrow7\leftarrow8$
Sub.	[16]	2.00	2.02	1.98	2.46	2.50	2.44	2.56	1.82	2.04
	[21]	3.06	3.00	3.02	3.34	3.34	3.32	3.24	3.26	2.98
	[18]	3.76	3.70	3.70	3.86	3.92	3.98	3.88	3.88	3.60
	VSRS	1.16	1.24	1.20	1.10	1.18	1.52	4.04	**4.06**	2.80
	Pro.	**4.52**	**4.54**	**4.52**	**4.52**	**4.64**	**4.66**	**4.06**	3.96	**3.80**

disocclusions are reconstructed through the polynomial fitting strategy. The proposed CC-IB effectively removes the color mismatches between the two side views. Lastly, a D-HHF method is put forward for filling the remaining holes. In which, holes are divided into two kinds under the guidance of depth, and different down-sampling strategies are introduced on these two kinds of holes. The proposed D-HHF ensures a maximum utilizations of the background pixels, so that the holes are filled reasonably and accurately. Experimental results demonstrate that the proposed method not only provides natural-looking virtual views, but also shows great advantages in quantitative evaluations. The future work is mainly about generating virtual view based on one-side view, besides, realizing real-time free viewpoint generation with graphics processing units (GPUs) is also the working orientation[41][42].

V. ACKNOWLEDGMENT

The authors would like to thank Microsoft Research, Nagoya, and ETRI/MPEG Korea Forum for distributing the multi-

camera sequences, which allow the authors to use the resources for research. The authors would like to thank the Graphics and Media Lab of Moscow State University for providing the Perceptual Video Quality tool, which allow the authors to evaluate the sequences conveniently. The authors also would like to thank the anonymous reviewers for their valuable comments and constructive suggestions for this paper.

REFERENCES

[1] A. Kubota, A. Smolic, M. Magnor, and M. Tanimoto, "Multiview imaging and 3dtv," *Signal Processing Magazine IEEE*, vol. 24, no. 6, pp. 10–21, 2007.

[2] M. Tanimoto and M. Wildeboer, "Frameworks for ftv coding," in *Picture Coding Symposium*, 2009, pp. 1–4.

[3] H. T. Quan, M. Barkowsky, and P. L. Callet, "The importance of visual attention in improving the 3d-tv viewing experience: Overview and new perspectives," *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 421–431, 2011.

[4] M. Tanimoto, "Ftv and all-around 3dtv," in *Visual Communications and Image Processing*, 2011, pp. 1–4.

[5] A. Hilton, J. Y. Guillemaut, J. Kilner, O. Grau, and G. Thomas, "3d-tv production from conventional cameras for sports broadcast," *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 462–476, 2011.

[6] W. Li, J. Zhou, B. Li, and M. I. Sezan, "Virtual view specification and synthesis for free viewpoint television," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 4, pp. 533–546, 2009.

[7] A. Smolic, *3D video and free viewpoint video-From capture to display*. Elsevier Science Inc., 2011.

[8] C. Lipski, F. Klose, and M. Magnor, "Correspondence and depth-image based rendering a hybrid approach for free-viewpoint video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 6, pp. 942–951, 2014.

[9] M. Tanimoto, "Ftv: Free-viewpoint television," *Signal Processing Image Communication*, vol. 27, no. 6, pp. 555–570, 2012.

[10] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W. T. Tan, "Loss-resilient coding of texture and depth for free-viewpoint video conferencing," *IEEE Transactions on Multimedia*, vol. 16, no. 3, pp. 711–725, 2014.

[11] J. Lei, C. Zhang, Y. Fang, Z. Gu, N. Ling, and C. Hou, "Depth sensation enhancement for multiple virtual view rendering," *IEEE Transactions on Multimedia*, vol. 17, no. 4, pp. 457–469, 2015.

[12] H. R. Kaviani and S. Shirani, "An adaptive patch-based reconstruction scheme for view synthesis by disparity estimation using optical flow," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. PP, no. 99, pp. 1–1, 2017.

[13] L. Wang and C. Jung, "Example-based video stereolization with foreground segmentation and depth propagation," *IEEE Transactions on Multimedia*, vol. 16, no. 7, pp. 1905–1914, 2014.

[14] J. Li, M. Barkowsky, and P. L. Callet, *Visual Discomfort in 3DTV: Definitions, Causes, Measurement, and Modeling*. Springer New York, 2015.

[15] Y. Zhao, C. Zhu, Z. Chen, and L. Yu, "Depth no-synthesis-error model for view synthesis in 3-d video," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2221–2228, Aug 2011.

[16] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, "View generation with 3d warping using depth information for ftv," in *3dtv Conference: the True Vision - Capture, Transmission and Display of 3d Video*, 2008, pp. 65–72.

[17] L. Do, S. Zinger, Y. Morvan, and P. H. N. D. With, "Quality improving techniques in dibr for free-viewpoint video," in *3dtv Conference: the True Vision - Capture, Transmission and Display of 3d Video*, 2009, pp. 1–4.

[18] S. Zinger, L. Do, and P. H. N. D. With, "Free-viewpoint depth image based rendering," *Journal of Visual Communication and Image Representation*, vol. 21, no. 5C6, pp. 533–541, 2010.

[19] W. J. Tam, F. Speranza, S. Yano, K. Shimono, and H. Ono, "Stereoscopic 3d-tv: Visual comfort," *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 335–346, 2011.

[20] D. Tian, P. L. Lai, P. Lopez, and C. Gomila, "View synthesis techniques for 3d video," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 7443, pp. 74 430T–74 430T–11, 2009.

[21] K. Oh, S. Yea, A. Vetro, and Y. Ho, "Virtual view synthesis method and self-evaluation metrics for free viewpoint television and 3d video," *International Journal of Imaging Systems and Technology*, vol. 20, no. 4, pp. 378–390, 2010.

[22] C. Yao, T. Tillo, Y. Zhao, J. Xiao, H. Bai, and C. Lin, "Depth map driven hole filling algorithm exploiting temporal correlation information," *IEEE Transactions on Broadcasting*, vol. 60, no. 2, pp. 394–404, 2014.

[23] W. Sun, O. C. Au, L. Xu, Y. Li, and W. Hu, "Novel temporal domain hole filling based on background modeling for view synthesis," in *IEEE International Conference on Image Processing*, 2012, pp. 2721–2724.

[24] C. Zhu and S. Li, "Depth image based view synthesis: New insights and perspectives on hole generation and filling," *IEEE Transactions on Broadcasting*, vol. 62, no. 1, pp. 82–93, 2016.

[25] I. Ahn and C. Kim, "A novel depth-based virtual view synthesis method for free viewpoint video," *IEEE Transactions on Broadcasting*, vol. 59, no. 4, pp. 614–626, Dec 2013.

[26] T. Zarb and C. J. Debono, "Depth-based image processing for 3d video rendering applications," in *International Conference on Systems, Signals and Image Processing*, 2014, pp. 215–218.

[27] U. Fecker, M. Barkowsky, and A. Kaup, "Improving the prediction efficiency for multi-view video coding using histogram matching," *in Picture Coding Symposium (PCS 2006*, 2006.

[28] X. Yang, J. Liu, J. Sun, X. Li, W. Liu, and Y. Gao, "Dibr based view synthesis for free-viewpoint television," in *3dtv Conference: the True Vision - Capture, Transmission and Display of 3d Video*, 2011, pp. 1–4.

[29] A. Telea, "An image inpainting technique based on the fast marching method," *Journal of Graphics Tools*, vol. 9, no. 1, pp. 23–34, 2004.

[30] M. Solh and G. AlRegib, "Hierarchical hole-filling(hhf): Depth image based rendering without depth map filtering for 3d-tv," in *2010 IEEE International Workshop on Multimedia Signal Processing*, Oct 2010, pp. 87–92.

[31] Y. Zhao, C. Zhu, Z. Chen, D. Tian, and L. Yu, "Boundary artifact reduction in view synthesis of 3d video: From perspective of texture-depth alignment," *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 510–522, June 2011.

[32] J. L. Barron and D. J. Fleet, "Beauchemin performance of optical flow techniques in international journal of computer vision," 1996.

[33] Y. Chen, C. Cai, and J. Liu, "Yuv correction for multi-view video compression," in *International Conference on Pattern Recognition*, 2006, pp. 734–737.

[34] V. Cheung, S. Westland, D. Connah, and C. Ripamonti, "A comparative study of the characterisation of color cameras by means of neural networks and polynomial transforms," vol. 120, pp. 19 – 25, 01 2004.

[35] F. Shao, G. Y. Jiang, M. Yu, and Y. S. Ho, "Fast color correction for multi-view video by modeling spatio-temporal variation," *Journal of Visual Communication and Image Representation*, vol. 21, no. 5-6, pp. 392–403, 2010.

[36] C. Doutre and P. Nasiopoulos, "Color correction preprocessing for multiview video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 9, pp. 1400–1406, 2009.

[37] MicrosoftResearch, "Image-basedrealities-3dvideodownload," http://research.microsoft.com/ivm/3DVideoDownload/S.

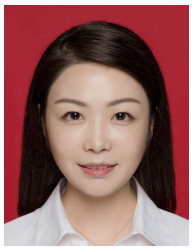
[38] L. Zhang, C. Vazquez, and S. Knorr, "3d-tv content creation: Automatic 2d-to-3d video conversion," *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 372–383, 2011.

[39] O. Stankiewicz, K. Wegner, M. Tanimoto, and M. Domaski, "Enhanced view synthesis reference software (vsrs) for free-viewpoint television," 01 2013.

[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, vol. 13, no. 4, p. 600, 2004.

[41] K. Suzuki, N. Fukushima, T. Yendo, M. P. Tehrani, T. Fujii, and M. Tanimoto, "Parallel processing method for realtime ftv," in *Picture Coding Symposium*, 2010, pp. 330–333.

[42] L. Do, G. Bravo, S. Zinger, and P. H. N. D. With, "Gpu-accelerated real-time free-viewpoint dibr for 3dtv," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 633–640, 2012.



Yiguo Qiao received the B.S. degree from Xidian University, Xian, China, in 2010. She is currently pursuing the Ph.D. degree at the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China, Xian, China. Her research interests include depth map up-sampling, free-viewpoint generation, and some other aspects in 3D stereoscopic vision, computer vision and machine learning.

Licheng Jiao (SM89) received the B.S. degree from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees from Xian Jiaotong University, Xian, China, in 1984 and 1990, respectively. Currently, he is a Professor and Dean with the Electronic Engineering School, Xidian University, Xian, China. His research interests include neural networks, data mining, nonlinear intelligence signal processing, and communication.

Shuyuan Yang received the B.S. degree in electrical engineering and the M.S. and Ph.D. degrees in circuit and system from Xidian University, Xian, China, in 2000, 2003, and 2005, respectively. She is currently a Professor with the School of Electronic Engineering, Xidian University. Her current research interests include compress sensing, machine learning, and intelligent information processing.

Biao Hou received the B.S. and M.S. degrees in mathematics from Northwest University, Xian, China, in 1996 and 1999, respectively, and the Ph.D. degree in circuits and systems from Xidian University, Xian, in 2003. Since 2003, he has been with the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, Xidian University, where he is currently a Professor. His research interests include compressive sensing and Synthetic Aperture Radar image interpretation.

Jie Feng received the B.S. degree from Changan University, Xian, China, in 2008. She is currently pursuing the Ph.D. degree at the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, School of Electronic Engineering, Xidian University, Xian, China. Her research interests include synthetic aperture radar image interpretation, hyperspectral image processing, pattern recognition, and evolutionary computation.